

PODSTAWY PROGRAMOWANIA W PYTHON

Dzień 3





AGENDA DAY 3

- Instrukcje warunkowe
- Code style



1. Instrukcje warunkowe

if - elif - else



BLOK KODU

Indentacja 1 poziom (4 spacje)

Instrukcja/wyrażenie:

Instrukcja

Instrukcja

Instrukcja:

instrukcja

Instrukcja/wyrażenie:

Instrukcja

Instrukcja

Instrukcja

Dwukropek rozpoczynający blok



PSEUDOKOD

- 1. weź książkę telefoniczną
- otwórz książkę na środku
- 3. zobacz nazwiska
- 4. jeśli "Wojtkowiak" jest wśród osób
- 5. zadzwoń do niego
- 6. w przeciwnym razie jeśli "Wojtkowiak" jest wcześniej w książce
- 7. otwórz lewą połowę po środku
- 8. Idź do kroku 3
- 9. w przeciwnym razie jeśli "Wojtkowiak" jest później w książce
- 10. otwórz prawą połowę po środku
- 11. idź do kroku 3
- 12. w przeciwnym razie
- 13. poddaj się

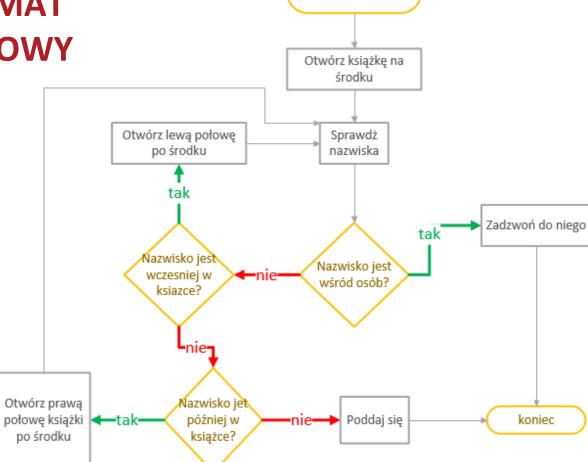


Instrukcje warunkowe

- 1. weź książkę telefoniczną
- 2. otwórz książkę na środku
- 3. zobacz nazwiska
- 4. jeśli "Wojtkowiak" jest wśród osób
- 5. zadzwoń do niego
- 6. w przeciwnym razie jeśli "Wojtkowiak" jest wcześniej w książce
- 7. otwórz lewą połowę po środku
- 8. Idź do kroku 3
- 9. w przeciwnym razie jeśli "Wojtkowiak" jest później w książce
- 10. otwórz prawą połowę po środku
- 11. idź do kroku 3
- 12. w przeciwnym razie
- 13. poddaj się



SCHEMAT BLOKOWY



Weź książkę

if-elif-else



```
if (warunek):
    # jakiś kod wykonany gdy warunek prawdziwy
elif (inny warunek):
    # kod wykonany gdy warunek w if był fałszywy
    # warunek w tym elif musi być prawdziwy aby ten kod wykonać
elif (inny warunek):
    # elif-ów może być wielu. lub żadnego, kod wew. elif
    # wykona się tylko gdy wszystkie wyższe warunki były fałszywe
else:
    # przypadek domyślny, tu nie sprawdzamy warunku, kod w else
    # będzie wykonany gdy wszystkie w if- elif były fałszywe
    # else może być tylko jeden lub wcale
```

Tablica logiczna



| Α | В | A and B | A or B |
|-------|-------|---------|--------|
| True | True | True | True |
| True | False | False | True |
| False | True | False | True |
| False | False | False | False |



2. Code Style



ZEN OF PYTHON import this

Beautiful is better than ugly.

Explicit is better than implicit.

Simple is better than complex.

Complex is better than complicated.

Flat is better than nested.

Sparse is better than dense.

Readability counts





Python style guide: https://www.python.org/dev/peps/pep-0008/





Thanks!!